

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	2465	(primary adj storage) and (secondary adj storage)	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:05
L2	8387591	copy\$5 or mirror\$4 or back\$1up	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:05
L3	1477	1 and 2	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:05
L4	933760	record or directory or log	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:06
L5	876	3 and 4	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:06
L6	505	5 and map\$5	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:06
L7	9	reconcile adj command	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:06
L8	2	6 and 7	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:07
L9	195	6 and reset\$4	US_PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:07

L10	134	9 and switch	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:17
L11	4	9 and (virtual adj switch)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/21 09:18


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)
**Search:**  The ACM Digital Library  The Guide

 writing first data to primary and second data to secondary stor

**THE ACM DIGITAL LIBRARY**
[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

writing first data to primary and second data to secondary storage

Found 114,239 of 166,953

Sort results  
by
 relevance

 Save results to a Binder

[Try an Advanced Search](#)
Display  
results
 expanded form

 [Search Tips](#)
[Try this search in The ACM Guide](#)
 Open results in a new  
window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale

### 1 RAID: high-performance, reliable secondary storage

Peter M. Chen, Edward K. Lee, Garth A. Gibson, Randy H. Katz, David A. Patterson  
 June 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 2

**Publisher:** ACM PressFull text available: [pdf\(3.60 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Disk arrays were proposed in the 1980s as a way to use parallelism between multiple disks to improve aggregate I/O performance. Today they appear in the product lines of most major computer manufacturers. This article gives a comprehensive overview of disk arrays and provides a framework in which to organize current and future work. First, the article introduces disk technology and reviews the driving forces that have popularized disk arrays: performance and reliability. It discusses the tw ...

**Keywords:** RAID, disk array, parallel I/O, redundancy, storage, striping

### 2 Comparison of access methods for time-evolving data

Betty Salzberg, Vassilis J. Tsotras  
 June 1999 **ACM Computing Surveys (CSUR)**, Volume 31 Issue 2

**Publisher:** ACM PressFull text available: [pdf\(529.53 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper compares different indexing techniques proposed for supporting efficient access to temporal data. The comparison is based on a collection of important performance criteria, including the space consumed, update processing, and query time for representative queries. The comparison is based on worst-case analysis, hence no assumptions on data distribution or query frequencies are made. When a number of methods have the same asymptotic worst-case behavior, features in the methods tha ...

**Keywords:** I/O performance, access methods, structures, temporal databases

### 3 Deconstructing Commodity Storage Clusters

Haryadi S. Gunawi, Nitin Agrawal, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Jiri Schindler

**June 2005 Proceedings of the 32nd Annual International Symposium on Computer Architecture ISCA '05**

**Publisher:** IEEE Computer Society

Full text available:  [pdf\(269.90 KB\)](#) Additional Information: [full citation](#), [abstract](#)

The traditional approach for characterizing complex systems is to run standard workloads and measure the resulting performance as seen by the end user. However, unique opportunities exist when characterizing a system that is itself constructed from standardized components: one can also look inside the system itself by instrumenting each of the components. In this paper, we show how intra-box instrumentation can help one understand the behavior of a large-scale storage cluster, the EMC Centera. I ...

**4 Fortran 8X draft**



 Loren P. Meissner

December 1989 **ACM SIGPLAN Fortran Forum**, Volume 8 Issue 4

**Publisher:** ACM Press

Full text available:  [pdf\(21.36 MB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

**Standard Programming Language Fortran.** This standard specifies the form and establishes the interpretation of programs expressed in the Fortran language. It consists of the specification of the language Fortran. No subsets are specified in this standard. The previous standard, commonly known as "FORTRAN 77", is entirely contained within this standard, known as "Fortran 8x". Therefore, any standard-conforming FORTRAN 77 program is standard conforming under this standard. New features can b ...

**5 Integrated document caching and prefetching in storage hierarchies based on**



Markov-chain predictions

Achim Kraiss, Gerhard Weikum

August 1998 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 7 Issue 3

**Publisher:** Springer-Verlag New York, Inc.

Full text available:  [pdf\(603.01 KB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

Large multimedia document archives may hold a major fraction of their data in tertiary storage libraries for cost reasons. This paper develops an integrated approach to the vertical data migration between the tertiary, secondary, and primary storage in that it reconciles speculative prefetching, to mask the high latency of the tertiary storage, with the replacement policy of the document caches at the secondary and primary storage level, and also considers the interaction of these policies with ...

**Keywords:** Caching, Markov chains, Performance, Prefetching, Scheduling, Stochastic modeling, Tertiary storage

**6 External memory algorithms and data structures: dealing with massive data**



 Jeffrey Scott Vitter

June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2

**Publisher:** ACM Press

Full text available:  [pdf\(828.46 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

**Keywords:** B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

## 7 Algorithms and data structures for flash memories

 Eran Gal, Sivan Toledo  
June 2005 **ACM Computing Surveys (CSUR)**, Volume 37 Issue 2

**Publisher:** ACM Press

Full text available:  pdf(343.39 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Flash memory is a type of electrically-erasable programmable read-only memory (EEPROM). Because flash memories are nonvolatile and relatively dense, they are now used to store files and other persistent objects in handheld computers, mobile phones, digital cameras, portable music players, and many other computer systems in which magnetic disks are inappropriate. Flash, like earlier EEPROM devices, suffers from two limitations. First, bits can only be cleared by erasing a large block of memory. S ...

**Keywords:** EEPROM memory, Flash memory, wear leveling

## 8 File servers for network-based distributed systems

 Liba Svobodova  
December 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 4

**Publisher:** ACM Press

Full text available:  pdf(4.23 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)

## 9 Consistent and automatic replica regeneration

 Haifeng Yu, Amin Vahdat  
February 2005 **ACM Transactions on Storage (TOS)**, Volume 1 Issue 1

**Publisher:** ACM Press

Full text available:  pdf(372.24 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Reducing management costs and improving the availability of large-scale distributed systems require automatic replica *regeneration*, that is, creating new replicas in response to replica failures. A major challenge to regeneration is maintaining consistency when the replica group changes. Doing so is particularly difficult across the wide area where failure detection is complicated by network congestion and node overload. In this context, this article presents Om, the first read/write peer- ...

**Keywords:** Peer-to-peer storage systems, availability, consistency, regeneration, replication

## 10 Access methods for multiversion data

 David Lomet, Betty Salzberg  
June 1989 **ACM SIGMOD Record , Proceedings of the 1989 ACM SIGMOD international conference on Management of data SIGMOD '89**, Volume 18 Issue 2

**Publisher:** ACM Press

Full text available:  pdf(1.11 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We present an access method designed to provide a single integrated index structure for

a versioned timestamped database with a non-deletion policy. Historical data (superceded versions) is stored separately from current data. Our access method is called the Time-Split B-tree. It is an index structure based on Malcolm Easton's Write Once B-tree. The Write Once B-tree was developed for data stored entirely on a Write-Once Read-Many or WORM optical ...

### 11 Distributed, object-based programming systems



 Roger S. Chin, Samuel T. Chanson  
March 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 1

**Publisher:** ACM Press

Full text available:  pdf(2.97 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The development of distributed operating systems and object-based programming languages makes possible an environment in which programs consisting of a set of interacting modules, or objects, may execute concurrently on a collection of loosely coupled processors. An object-based programming language encourages a methodology for designing and creating a program as a set of autonomous components, whereas a distributed operating system permits a collection of workstations or personal computers ...

**Keywords:** capability scheme, distributed operating systems, error recovery, method invocation, nested transaction, object model, object reliability, object-based programming languages, processor allocation, resource management, synchronization, transaction

### 12 AMESPILOT—a higher level data plotting software system



 Ian Hirschsohn  
September 1970 **Communications of the ACM**, Volume 13 Issue 9

**Publisher:** ACM Press

Full text available:  pdf(1.79 MB)

Additional Information: [full citation](#), [references](#), [citations](#)

**Keywords:** computer graphics, data display syntax, data plotting, display device independent software, hardware independent software, map display, plot elements, projection, self-scaled plots, tablet organization, user interaction

### 13 Private information storage (extended abstract)



 Rafail Ostrovsky, Victor Shoup  
May 1997 **Proceedings of the twenty-ninth annual ACM symposium on Theory of computing**

**Publisher:** ACM Press

Full text available:  pdf(1.33 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

### 14 File and storage systems: The Google file system



 Sanjay Ghemawat, Howard Gobioff, Shun-Tak Leung  
October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

**Publisher:** ACM Press

Full text available:  pdf(275.54 KB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Keywords:** clustered storage, data storage, fault tolerance, scalability

**15 Distributed file systems: concepts and examples**



Eliezer Levy, Abraham Silberschatz  
December 1990 **ACM Computing Surveys (CSUR)**, Volume 22 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(5.33 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The purpose of a distributed file system (DFS) is to allow users of physically distributed computers to share data and storage resources by using a common file system. A typical configuration for a DFS is a collection of workstations and mainframes connected by a local area network (LAN). A DFS is implemented as part of the operating system of each of the connected computers. This paper establishes a viewpoint that emphasizes the dispersed structure and decentralization of both data and con ...

**16 OceanStore: an architecture for global-scale persistent storage**



John Kubiatowicz, David Bindel, Yan Chen, Steven Czerwinski, Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Chris Wells, Ben Zhao  
November 2000 **ACM SIGARCH Computer Architecture News , ACM SIGOPS Operating Systems Review , Proceedings of the ninth international conference on Architectural support for programming languages and operating systems ASPLOS-IX**, Volume 28 , 34 Issue 5 , 5

**Publisher:** ACM Press

Full text available: [pdf\(166.53 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

OceanStore is a utility infrastructure designed to span the globe and provide continuous access to persistent information. Since this infrastructure is comprised of untrusted servers, data is protected through redundancy and cryptographic techniques. To improve performance, data is allowed to be cached anywhere, anytime. Additionally, monitoring of usage patterns allows adaptation to regional outages and denial of service attacks; monitoring also enhances performance through pro-active movement ...

**17 Concepts and capabilities of a database computer**



Jayanta Banerjee, David K. Hsiao, Richard I. Baum  
December 1978 **ACM Transactions on Database Systems (TODS)**, Volume 3 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(2.79 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The concepts and capabilities of a database computer (DBC) are given in this paper. The proposed design overcomes many of the traditional problems of database system software and is one of the first to describe a complete data-secure computer capable of handling large databases. This paper begins by characterizing the major problems facing today's database system designers. These problems are intrinsically related to the nature of conventional hardware and can only be solved by i ...

**Keywords:** clustering, content-addressable memory, database computers, keywords, mass memory, performance, security, structure memory

**18 OceanStore: an architecture for global-scale persistent storage**



John Kubiatowicz, David Bindel, Yan Chen, Steven Czerwinski, Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Westley Weimer, Chris Wells, Ben Zhao

November 2000 **ACM SIGPLAN Notices**, Volume 35 Issue 11

**Publisher:** ACM Press

Full text available:  [pdf\(1.47 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

OceanStore is a utility infrastructure designed to span the globe and provide continuous access to persistent information. Since this infrastructure is comprised of untrusted servers, data is protected through redundancy and cryptographic techniques. To improve performance, data is allowed to be cached anywhere, anytime. Additionally, monitoring of usage patterns allows adaptation to regional outages and denial of service attacks; monitoring also enhances performance through pro-active movement ...

**19** Petal: distributed virtual disks 

 Edward K. Lee, Chandramohan A. Thekkath

September 1996 **ACM SIGPLAN Notices , ACM SIGOPS Operating Systems Review , Proceedings of the seventh international conference on Architectural support for programming languages and operating systems ASPLOS-VII**, Volume 31 , 30 Issue 9 , 5

**Publisher:** ACM Press

Full text available:  [pdf\(1.10 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The ideal storage system is globally accessible, always available, provides unlimited performance and capacity for a large number of clients, and requires no management. This paper describes the design, implementation, and performance of Petal, a system that attempts to approximate this ideal in practice through a novel combination of features. Petal consists of a collection of network-connected servers that cooperatively manage a pool of physical disks. To a Petal client, this collection appear ...

**20** Proceedings of the SIGNUM conference on the programming environment for 

 development of numerical software

March 1979 **ACM SIGNUM Newsletter**, Volume 14 Issue 1

**Publisher:** ACM Press

Full text available:  [pdf\(5.02 MB\)](#) Additional Information: [full citation](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)




[Home](#) | [Login](#) | [Logout](#) | [Access Information](#)

Welcome United States Patent and Trademark Office

[Advanced Search](#)[BROWSE](#)[SEARCH](#)[IEEE Xplore GUIDE](#)**OPTION 1**

Enter keywords or phrases, select fields, and select operators

 In All Fields

 AND  In All Fields

 AND  In All Fields
[Help](#)

» Note: If you use all three search boxes, the entries in the first two boxes take precedence over the entry in the third box.

**OPTION 2**

Enter keywords, phrases, or a Boolean expression

```
(primary<near/1>storage)<and>
(secondry<near/1>storage)<and>
(reconcile<near/1>switch)
```

[Help](#)

» Note: You may use the search operators <and> or <or> without the start and end brackets <>.

» Learn more about [Field Codes](#), [Search Examples](#), and [Search Operators](#)

## » Publications

 Select publications IEEE Periodicals IEE Periodicals IEEE Conference IEE Conference P IEEE Standards

## » Other Resources (Available)

 IEEE Books

## » Select date range

 Search latest content From year to 

## » Display Format

 Citation Citation

## » Organize results

Maximum Display  resultsSort by In [Help](#) [Contact Us](#)

© Copyright 2005

 Indexed by